

(1)	62	DECLARATIONS
(2)	89	ADD TO WORKING SET SERVICE
(3)	216	WSPEAKCHK - ENABLE OR DISABLE WORKING SET PEAK CHECKING
(4)	256	SHRINK WORKING SET
(5)	379	EXTRADYNWS - CALCULATE EXTRA DYNAMIC WORKING SET COUNT

SYS
Synd

ACM
CTL
CTL
CTL
EXE
EXE
IPL
IPL
MMG
MMG
MMG
MMG
PAG
PCB
PCB
PCB
PFN
PHD
PR
SCH
SCH
SCH
SCH
SGN
SGN
SHR
SSS
SSS
HSE

PSE

SAB
YSE
SMM

0000 1 .TITLE SYSADJWSL - SYSTEM SERVICE ADJUST WORKING SET LIMIT
0000 2 .IDENT 'V04-000'
0000 3 .
0000 4 .
0000 5 .*****
0000 6 .*: COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 7 .*: DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 8 .*: ALL RIGHTS RESERVED.
0000 9 .
0000 10 .*: THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 11 .*: ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 12 .*: INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 13 .*: COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 14 .*: OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 15 .*: TRANSFERRED.
0000 16 .
0000 17 .*: THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 18 .*: AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 19 .*: CORPORATION.
0000 20 .
0000 21 .*: DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 22 .*: SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 23 .
0000 24 .
0000 25 .
0000 26 .*****
0000 27 .
0000 28 .++
0000 29 .: FACILITY: EXECUTIVE, MEMORY MANAGEMENT SERVICE
0000 30 .
0000 31 .: ABSTRACT: SYSADJWSL IMPLEMENTS THE ADJUST WORKING SET LIMIT
0000 32 .: SYSTEM SERVICE.
0000 33 .
0000 34 .: ENVIRONMENT:
0000 35 .
0000 36 .: AUTHOR: PETER H. LIPMAN , CREATION DATE: 10-DEC-76
0000 37 .
0000 38 .: MODIFIED BY:
0000 39 .
0000 40 .: V03-005 WMC0003 Wayne Cardoza 05-MAY-1983
0000 41 .: Change ESTRADYNWS calculation to account for locked page
0000 42 .: tables in the dynamic portion of the working set.
0000 43 .
0000 44 .: V03-004 TCM0001 Trudy C. Matthews 1-Apr-1983
0000 45 .: Change references to working set fields in PHD so that
0000 46 .: they are used as unsigned words.
0000 47 .
0000 48 .: V03-003 WMC0002 Wayne Cardoza 22-Dec-1982
0000 49 .: Expansion of process header for new working set entries must
0000 50 .: be iterative process. Don't request more than can currently
0000 51 .: be locked in working set.
0000 52 .
0000 53 .: V03-002 WMC0001 Wayne Cardoza 04-Oct-1982
0000 54 .: EXTRADYNNS changed to use minimum of quota and size.
0000 55 .
0000 56 .: V03-001 HRJ0062 Herb Jacobs 05-Apr-1982
0000 57 .: Fix ESTRADYNWS calculation. This could cause IPL 2 loops

- SYSTEM SERVICE ADJUST WORKING SET LIMI 16-SEP-1984 01:38:14 VAX/VMS Macro V04-00
D 11
5-SEP-1984 03:48:37 [SYS.SRC]SYSADJWSL.MAR;1

Page 2
(1)

**F

0000 58 :
0000 59 :
0000 60 :--

by being too lenient.

```
0000 62 .SBTTL DECLARATIONS
0000 63 ; INCLUDE FILES:
0000 64 ; SACMDEF :ACCOUNTING MESSAGE DEFINITIONS
0000 65 ; SIPLDEF :PROCESSOR PRIORITY LEVELS
0000 66 ; SPCBDEF :PROCESS CONTROL BLOCK DEFINITIONS
0000 67 ; SPHDDF :PROCESS HEADER DEFINTIONS
0000 68 ; SPRDEF :PROCESSOR REGISTER DEFINITIONS
0000 69 ; SPTEDF :PAGE TABLE ENTRY DEFINITIONS
0000 70 ; SSECDEF :SECTION TABLE OFFSET DEFINTIONS
0000 71 ; SSSDEF :SYSTEM SERVICE DEFINITIONS
0000 72 ;
0000 73 ;
0000 74 ; MACROS:
0000 75 ;
0000 76 ;
0000 77 ;
0000 78 ; EQUATED SYMBOLS:
0000 79 ;
0000 80 ; OFFSETS FROM AP
0000 81 ;
0000 82 ; PAGCNT = 4 :NUMBER OF PAGES TO ADD OR SUBTRACT
00000004 0000 83 ; WSETLM = 8 :ADDRESS TO RETURN NEW WORKING SET LIMIT
00000008 0000 84 ;
0000 85 ;
0000 86 ; OWN STORAGE:
0000 87 ;
```

0000 89 .SBTTL ADD TO WORKING SET SERVICE
 0000 90 ++
 0000 91 FUNCTIONAL DESCRIPTION:
 0000 92
 0000 93
 0000 94 CALLING SEQUENCE:
 0000 95 CALL ARGLIST,SYSSADJWSL
 0000 96
 0000 97 INPUT PARAMETERS:
 0000 98
 0000 99
 100 100 PAGCNT(AP) = NUMBER OF PAGES TO ADD IF POSITIVE
 100 101 = NUMBER OF PAGES TO SUBTRACT IF NEGATIVE
 100 102 WSETLM(AP) = ADDRESS OF LONG WORD TO RETURN NEW WORKING SET LIMIT
 100 103 R4 = PROCESS CONTROL BLOCK ADDRESS
 100 104 SET UP BY THE CHANGE MODE TO KERNEL DISPATCHER
 100 105
 106 106 IMPLICIT INPUTS:
 106 107
 108 108 NONE
 108 109
 110 110 OUTPUT PARAMETERS:
 110 111 R0 = SYSTEM STATUS CODE
 110 112
 114 114 IMPLICIT OUTPUTS:
 114 115
 116 116 NONE
 116 117
 118 118 COMPLETION CODES:
 118 119
 120 120 NONE
 120 121
 122 122 SIDE EFFECTS:
 122 123
 124 124 NONE
 124 125
 126 126 --
 126 127
 128 128 *****
 128 129 ***** THE FOLLOWING CODE MAY BE PAGED *****
 128 130 *****
 131 131 .PSECT YSEXEPAGED
 131 132
 132 133
 133 134 *****
 134 135 *****
 135 136 EXE\$ADJWSL:
 003C 136 WORD ^M<R2,R3,R4,R5>
 0002 137 SETIPL #IPL\$ AS\$DEL :NO AST'S WHILE MODIFYING PHD
 0005 138 MOVL G^CTL\$GL PHD,R5 :PROCESS HEADER ADDRESS (P1 SPACE)
 000C 139 MOVL PAGCNT(AP),R1 :NO. OF PAGES TO ADD TO WORKING SET
 0010 140 141 :**** REFERENCE COULD PAGE FAULT
 12 142 BGEQ 30\$:BRANCH IF GROW WORKING SET
 00000000'EF 143 JSB SHRINK :SHRINK THE WORKING SET
 00A4 144 BRW 60\$:EXIT SYSTEM SERVICE
 001B 145 ;

55 00000000'GF D0 0005 137 WORD ^M<R2,R3,R4,R5>
 51 04 AC D0 000C 138 SETIPL #IPL\$ AS\$DEL :NO AST'S WHILE MODIFYING PHD
 00000000'EF 12 18 0010 139 MOVL G^CTL\$GL PHD,R5 :PROCESS HEADER ADDRESS (P1 SPACE)
 00A4 16 0012 140 MOVL PAGCNT(AP),R1 :NO. OF PAGES TO ADD TO WORKING SET
 001B 31 0018 141 :**** REFERENCE COULD PAGE FAULT
 142 BGEQ 30\$:BRANCH IF GROW WORKING SET
 143 JSB SHRINK :SHRINK THE WORKING SET
 144 BRW 60\$:EXIT SYSTEM SERVICE
 145 ;

001B 146 : MINIMUM OF R1 AND R2 TO R1

52 51 D1 001B 147 10\$: CMPL R1, R2 ;USE THE SMALLER
03 15 001E 148 10\$: BLEQ 20\$;BRANCH IF R1 IS THE SMALLER
51 52 D0 0020 149 20\$: MOVL R2, R1 ;USE R2 INSTEAD
05 0023 150 151 152 153 30\$: MOVL L^SGN\$GL MAXWSCNT, R2 ;MAXIMUM SYSTEM WIDE WORKING SET SIZE
00000000'EF 0024 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 10\$: MOVL PHDSW_WSSIZE(R5), R0 ;CURRENT WORKING SET SIZE
50 50 A5 3C 002B 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: MOVL PHDSW_WSLIST(R5), R3 ;GET START OF WSL ENTRIES
53 08 A5 3C 002F 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: DECL R3 ;ADJUST TO GET COUNT WHEN SUBTRACTED
52 50 C2 0033 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: SUBL R0, R2 ;PAGES THAT POTENTIALLY MAY BE ADDED
52 16 A5 3C 003A 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: MOVZWL PHDSW_WSEXTENT(R5), R2 ;USE SMALLER OF REQUEST AND MAX
52 53 C2 003E 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: SUBL R3, R2 ;GET POINTER TO END OF MAX SIZE
52 50 C2 0041 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: SUBL R0, R2 ;GET MAXIMUM ALLOWED SIZE
52 50 C4 10 0044 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: BSBBL3 ;REDUCE BY WHAT'S IN USE
52 00000000'EF C3 0046 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 10\$: SUBL3 SCHSGL_FREELIM, - ;USE SMALLER OF REQUEST AND PROCESS MAX
52 00000000'EF 004C 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 170: PFNSGL_PHYPGCNT, R2 ;MAX PHYSICAL NUMBER OF PAGES
52 50 C2 0052 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 170: SUBL R0, R2 ;IS UPPER BOUND ON WORKING SET SIZE
52 50 C4 10 0055 166 167 168 169 170 171 172 173 174 175 176 177 178 179 170: BSBBL3 ;MAXIMUM NUMBER OF ADDITIONAL PAGES
52 12 A5 3C 0057 167 168 169 170 171 172 173 174 175 176 177 178 179 170: MOVZWL PHDSW_WSLAST(R5), R2 ;USE SMALLER
52 53 C2 005B 168 169 170 171 172 173 174 175 176 177 178 179 170: SUBL R3, R2 ;GET POINTER TO END OF CURRENT SIZE
52 50 C2 005E 169 170 171 172 173 174 175 176 177 178 179 170: SUBL R0, R2 ;NUMBER OF POTENTIAL PAGES
52 51 D1 0061 170 171 172 173 174 175 176 177 178 179 170: CMPL R1, R2 ;GET NUMBER OF UNUSED WSLE IN WSL
52 03 18 0064 171 172 173 174 175 176 177 178 179 170: BGEQ 40\$;IS REQUEST FOR LESS THAN UNUSED ONES?
52 51 D0 0066 172 173 174 175 176 177 178 179 170: MOVL R1, R2 ;BRANCH IF NOT
50 A5 52 A0 0069 173 40\$: ADDW R2, PHDSW_WSSIZE(R5) ;USE ONLY REQUEST SIZE
51 52 C2 006D 174 175 176 177 178 179 170: SUBL R2, R1 ;ADD IN THE FREE EXPANSION AMOUNT
4D 15 0070 175 176 177 178 179 170: BLEQ 60\$;NUMBER OF ADDITIONAL PAGES BEYOND WSLAST
51 51 DD 0072 176 177 178 179 170: PUSHL R1 ;BRANCH IF DONE
50 76 A5 3C 0074 177 178 179 170: MOVZWL PHDSW_EXTDYNWS(R5), R0 ;SAVE THE REQUEST SIZE
50 50 D6 0078 178 179 180 181 182 183 184 185 186 187 188 189 180: INCL R0 ;EXTRA WORKING SET ENTRIES
50 07 78 007A 179 180 181 182 183 184 185 186 187 188 189 180: ASHL #7, R0, R0 ;THERE IS ALWAYS AT LEAST ONE MORE THAN EXTR
50 51 D1 007E 180 181 182 183 184 185 186 187 188 189 180: CMPL R1, R0 ;LONGWORDS OF PROCESS HEADER THIS CAN LOCK
50 03 15 0081 181 182 183 184 185 186 187 188 189 180: BLEQ 45\$;CAN WE LOCK ENTIRE REQUEST
51 50 D0 0083 182 183 184 185 186 187 188 189 180: MOVL R0, R1 ;NO PROBLEM
00000000'EF 16 0086 183 45\$: JSB MMGSALCPHD ;REQUEST LESS
53 8E D0 008C 184 185 186 187 188 189 180: MOVL (SP)+, R3 ;GET ENOUGH SPACE FOR SPECIFIED # OF WSLE'S
008F 185: ;GET BACK THE REQUEST SIZE
008F 186: ; R1 IS RETURNED AS THE MINIMUM OF WHAT WAS REQUESTED AND WHAT WAS AVAILABLE
008F 187: ; NOTE THAT THE FOLLOWING CODE MUST WORK CORRECTLY IF NO NEW ENTRIES ARE ADDED
008F 188: ;
52 51 D0 008F 189 190 191 192 193 194 195 196 197 198 199 200 201 202 10\$: MOVL R1, R2 ;SAVE COUNT FOR AFTER LOOP
2B 13 0092 190 191 192 193 194 195 196 197 198 199 200 201 202 10\$: BEQL 60\$;BRANCH IF HEADER COULDN'T BE EXPANDED
50 12 A5 3C 0094 191 192 193 194 195 196 197 198 199 200 201 202 10\$: MOVZWL PHDSW_WSLAST(R5), R0 ;GET INDEX TO CURRENT END
50 10 A5 50 B0 0098 192 193 194 195 196 197 198 199 200 201 202 10\$: MOVW R0, PHDSW_WSNEXT(R5) ;UPDATE NEXT POINTER TO NEW FREE AREA
50 04 A540 DE 009C 193 194 195 196 197 198 199 200 201 202 10\$: MOVAL 4(R5)[R0], R0 ;GET POINTER TO NEW FIRST FREE ONE
80 D4 00A1 194 195 196 197 198 199 200 201 202 10\$: CLRL (R0)+ ;MARK ENTRY FREE
FB 51 F5 00A3 195 196 197 198 199 200 201 202 10\$: SOBGTR R1, 50\$;ONCE FOR EACH NEW WORKING SET ENTRY
12 A5 52 A0 00A6 196 197 198 199 200 201 202 10\$: ADDW R2, PHDSW_WSLAST(R5) ;UPDATE TO NEW WSLAST
50 A5 52 A0 00AA 197 198 199 200 201 202 10\$: ADDW R2, PHDSW_WSSIZE(R5) ;UPDATE TO NEW WORKING SET SIZE
53 52 D1 00AE 198 199 200 201 202 10\$: CMPL R2, R3 ;DO WE HAVE ALL WE NEEDED
0C 18 00B1 199 200 201 202 10\$: BGEQ 60\$;YES
000000C2'EF 16 00B3 200 201 202 10\$: JSB MMGSEXTADYNWS ;RECOMPUTE EXTRA DYNAMIC WSLE COUNT
51 53 52 C3 00B9 201 202 10\$: SUBL3 R2, R3, R1 ;AMOUNT WE STILL NEED
B3 11 00BD 202 10\$: BRB 42\$;GO TRY AGAIN

000000C2'EF	16	00BF	203	60\$:	JSB	MMGSEXTRADYNWS	:RECOMPUTE EXTRA DYNAMIC WSLE COUNT
16	10	00C5	204		BSBB	MMGSWSPEAKCHK	:ENABLE WORKING SET PEAK CHECK
52	08	AC	00C7	205	MOVL	WSETLM(AP),R2	:RETURN WORKING SET SIZE IN R1
50	0C	13	00CB	206	BEQL	80\$:ADDRESS TO RETURN NEW WORKING SET LIMIT
50	0C	3C	00CD	207	MOVZWL	#SSS_ACCVIO,RO	:***** COULD FAULT THIS REFERENCE
62	51	3C	00D6	208	IFNOWRT	#4,(R2),90\$:BRANCH IF NONE SPECIFIED
50	01	3C	00D9	209	MOVZWL	R1,(R2)	:RETURN CODE FOR ACCESS VIOLATION
				210			:IF NOT WRITABLE
				211			:OTHERWISE RETURN THE NEW SIZE
				212			:***** COULD FAULT THIS REFERENCE
				213	80\$:	MOVZWL #SSS_NORMAL,RO	:INDICATE SUCCESSFUL COMPLETION
				214	90\$:	RET	:AND RETURN


```

0108 256 .SBTTL SHRINK WORKING SET
0108 257 ****
0108 258 **** THE FOLLOWING CODE MUST BE RESIDENT ****
0108 259 ****
0108 260 ****
0108 261 ****
00000000 262 .PSECT SMMGCOD
0000 263 ****
0000 264 ****
0000 265 ****
0000 266 ****
0000 267 ****
0000 268 ****
0000 269 ****
0000 270 ****
0000 271 ****
0000 272 ****
0000 273 ****
0000 274 ****
0000 275 ****
0000 276 ****
0000 277 ****
0000 278 ****
0000 279 SHRINK:
7E 12 A5 3C 0003 280 SETIPL #IPL$_SYNCH :DISABLE SWAPPER
7E 50 A5 3C 0007 281 MOVZWL PHDSW_WSLAST(R5),-(SP) :SAVE IN CASE FREWSLE SHRINKS THIS
2C 10 000B 282 MOVZWL PHDSW_WSSIZE(R5),-(SP) :SAVE IN CASE RESOURCEWAIT NEEDED
0C BA 000D 283 BSBB MMG$SHRINKWS :SHRINK THE WORKING SET
04 12 000F 284 POPR #^M<R2,R3> :GET BACK ORIGINAL WSSIZE AND WSLAST
0011 285 BNEQ 10$ :BRANCH IF NON SUCCESSFUL
0014 286 SETIPL #IPL$_ASTDEL :SWAPPABLE AGAIN
0015 287 RSB
0015 288 : MUST WAIT FOR A RESOURCE
0015 289 : IPL = SYNCH, R1 = RESOURCE TO WAIT FOR
0015 290 : R4 = PROCESS CONTROL BLOCK ADDRESS
0015 291
0015 292 10$: MOVW R2,PHDSW_WSSIZE(R5) :RESET ORIGINAL WS SIZE FOR RETRY
12 A5 52 B0 0015 293 MOVW R3,PHDSW_WSLAST(R5) :RESET WSLAST, ENTRIES ARE KNOWN ZEROED
53 B0 0019 294 PUSHR #^M<R1> :SAVE RESOURCE WAIT CODE
02 BB 001D 295 BSBW MMG$EXTRADYNWS :RESET EXTRA DYNAMIC WORKING SET COUNT
00A0 30 001F 296 POPR #^M<R1> :RESTORE RESOURCE WAIT NUMBER
02 BA 0022 297 MOVL R1,PCBSL_EFWM(R4) :SET RESOURCE TO WAIT FOR
4C A4 51 D0 0024 298 BBSSI R1,W$CH$GL_RESMASK,20$ :NOTE SOMEONE WAITING
51 E6 0028 299 MOVAQ W$CH$GQ_MWAIT,R2 :WAIT ON MUTEXT WAIT QUEUE
00 0000'CF 51 7E 002E 300 20$: MOVL FP,SP :RESET FP, AP UNTOUCHED
52 0000'CF 51 D0 0033 301 BRW SCHSWAIT :WAIT AS CALLER
FFC7' 31 0036 302
0039 303
0039 304 MMG$SHRINKWS:: :SET DEFAULT RETURN STATUS
00 DD 0039 305 PUSHL #0 :GET A SCRATCH REGISTER
52 DD 003B 306 PUSHL R2 :KEEP WORKING SET SIZE IN R2
50 A5 3C 003D 307 MOVZWL PHDSW_WSSIZE(R5),R2 :MAKE THE DESIRED PAGE COUNT POSITIVE
53 51 CE 0041 308 MNEG R1,R3
0044 309 : CALCULATE THE MAXIMUM AMOUNT THE WORKING SET CAN BE REDUCED
0044 310
0044 311
50 0E A5 08 A5 A3 0044 312 SUBW3 PHDSW_WSLIST(R5),PHDSW_WSDYN(R5),R0 ;GET SIZE OF LOCKED WS

```

50 50 50 3C 004A 313 MOVZWL R0,R0 ;GET IT IN A LONGWORD
 51 52 50 C3 004D 314 SUBL3 R0,R2,R0 ;GET SIZE OF UNLOCKED WS (WSSIZE-LOCKED)
 50 0000'CF 3C 0051 315 MOVZWL W\$GN\$GW_MINWSCNT,R1 ;GET MINIMUM WS SIZE
 51 51 C2 0056 316 SUBL R1,R0 ;ALLOW CUSHION PAGES
 50 50 D7 0059 317 DECL R0 ;IN CASE CUSHION IS 0
 005B 318 :
 005B 319 : RO = NUMBER OF PAGES WHICH COULD BE TAKEN AWAY WITHOUT REDUCING
 005B 320 : THE WORKING SET SIZE BELOW THE MINIMUM.
 005B 321 : NOW CALCULATE AMOUNT IT CAN BE REDUCED WITHOUT DEPLETING EXTRADYNs
 005B 322 :
 51 18 A5 08 A5 A3 005B 323 SUBW3 PHDSW_WSLIST(R5),PHDSW_WSQUOTA(R5),R1
 51 51 B6 0061 324 INCW R1 ;QUOTA
 51 52 51 C3 0063 325 MOVZWL R1,R1 ;GET IT IN A LONGWORD
 51 06 15 0066 326 SUBL3 R1,R2,R1 ;AMOUNT SIZE EXCEEDS QUOTA
 51 76 A5 A0 006C 327 BLEQ 5\$;SIZE SMALLER - USE EXTRADYNs
 51 06 11 0070 328 ADDW PHDSW_EXTDYNWS(R5),R1 ;ADD IN THE EXCESS DYNAMIC ENTRIES
 51 76 A5 3C 0072 329 BRB 7\$
 43 13 0076 330 5\$: MOVZWL PHDSW_EXTDYNWS(R5),R1 ;EXCESS DYNAMIC WORKING SET LIST ENTRIES
 0078 331 BEQL 60\$;BRANCH IF NONE LEFT TO TAKE AWAY
 0078 332 :
 0078 333 : R1 = NUMBER OF EXTRA DYNAMIC WORKING SET LIST ENTRIES ABOVE
 0078 334 : THE MINIMUM REQUIRED BY WSFLUID.
 0078 335 :
 0078 336 :
 0078 337 : USE THE SMALLER OF RO AND R1 AS THE MOST PAGES THAT CAN BE TAKEN
 0078 338 : AWAY FROM THE WORKING SET LEAVING THE RESULT IN RO
 0078 339 :
 51 50 D1 0078 340 7\$: CMPL R0,R1
 50 03 15 007B 341 BLEQ 10\$
 50 51 D0 007D 342 MOVL R1,R0
 0080 343 10\$:
 0080 344 :
 0080 345 : RO IS THE MAXIMUM NUMBER OF PAGES THAT CAN BE TAKEN OUT OF THE WORKING SET
 0080 346 : REDUCE THE WORKING SET SIZE BY THE SMALLER OF RO AND R3.
 0080 347 :
 50 53 D1 0080 348 CMPL R3,R0 ;REDUCE BY THE DESIRED AMOUNT?
 05 05 15 0083 349 BLEQ 20\$;BRANCH IF YES
 53 50 D0 0085 350 MOVL R0,R3 ;MUST REDUCE BY LESS
 31 15 0088 351 BLEQ 60\$;BRANCH IF NO SHRINKING POSSIBLE
 008A 352 :
 008A 353 : NOW WE KNOW NUMBER OF PAGES WE CAN FREE, FIRST ATTEMPT TO RECOVER THE
 008A 354 : PAGES BY JUST REMOVING THE UNUSED GROWTH PAGES FROM THE WORKING SET.
 008A 355 : AFTER THIS PAGES WILL BE FREED BY USING THE NORMAL PAGEFAULT REPLACEMENT
 008A 356 : ALGORITHM.
 008A 357 :
 50 34 A4 36 A4 A1 008A 358 20\$: ADDW3 PCB\$W_PPGCNT(R4),PCB\$W_GPGCNT(R4),R0 ;PAGE COUNT CURRENTLY IN USE
 51 50 50 3C 0090 359 MOVZWL R0,R0 ;GET IT IN A LONGWORD
 52 50 C3 0093 360 SUBL3 R0,R2,R1 ;NUMBER OF PAGES IMMEDIATELY RECLAIMABLE
 53 11 15 0097 361 BLEQ 40\$;BRANCH IF NONE
 53 51 D1 0099 362 CMPL R1,R3 ;ARE WE GOING TO GET BACK TOO MANY?
 03 15 009C 363 BLEQ 30\$;BRANCH IF NOT
 51 53 D0 009E 364 MOVL R3,R1 ;TAKE BACK ONLY WHAT WAS ASKED
 50 A5 51 A2 00A1 365 30\$: SUBW R1,PHD\$W_WSSIZE(R5) ;ADJUST WORKING SET DOWN BY EMPTIES
 53 51 C2 00A5 366 SUBL R1,R3 ;ADJUST COUNT OF PAGES TO STILL FREE
 11 15 00A8 367 BLEQ 60\$;BRANCH IF DONE
 04 AE 53 D0 00AA 368 40\$: MOVL R3,4(SP) ;SAVE COUNT OF PAGES TO FREE
 00AA 369

FF4F' 30 00AE 370 50\$: ;
07 50 E9 00B1 371 BSBW MMGSFREWSLE ;
50 A5 B7 00B4 372 BLBC R0,60\$;
F3 04 AE F5 00B7 373 DECW PHDSW_WSSIZE(R5) ;
52 8E D0 00BB 374 SOBGTR 4(SP),50\$;
50 8E D0 00BE 375 60\$: MOVL (SP)+,R2 ;
05 00C1 376 MOVL (SP)+,R0 ;
RSB ;
;GET A FREE WORKING SET LIST ENTRY
;BRANCH TO RETURN WITH NON ZERO COUNT
;ACCOUNT FOR NEWLY FREED PAGE
;REPEAT FOR EACH SLOT TO BE DELETED
;RESTORE R2
;SET RETURN STATUS- NON-ZERO=FAILURE
;RETURN

00C2 379 .SBTTL EXTRADYNWS - CALCULATE EXTRA DYNAMIC WORKING SET COUNT
 00C2 380
 00C2 381
 00C2 382 : FUNCTIONAL DESCRIPTION:
 00C2 383 : THIS ROUTINE CALCULATES THE EXTRA NUMBER OF DYNAMIC WORKING SET
 00C2 384 : ENTRIES AVAILABLE BEYOND THE NUMBER GUARANTEED BY WSFLUID
 00C2 385 :
 00C2 386 : CALLING SEQUENCE:
 00C2 387 :
 00C2 388 :
 00C2 389 : BSBW MMGSEXTRADYNWS
 00C2 390 :
 00C2 391 : INPUTS:
 00C2 392 :
 00C2 393 : R5 = PROCESS HEADER ADDRESS
 00C2 394 : IPL = AT LEAST ASTDEL
 00C2 395 :
 00C2 396 : OUTPUTS:
 00C2 397 :
 00C2 398 : R1 = EXTRA DYNAMIC WORKING SET COUNT
 00C2 399 : R0, R2, R3 PRESERVED
 00C2 400 : PHDSW_EXTDYNWS = EXTRA DYNAMIC WORKING SET COUNT
 00C2 401 :
 00C2 402 :
 00C2 403 : ***** THE FOLLOWING CODE MUST BE RESIDENT *****
 00C2 404 :
 00C2 405 :
 000000C2 406 : .PSECT \$MMGCOD
 00C2 407 :
 00C2 408 :
 00C2 409 :
 00C2 410 :
 00C2 411 :
 00C2 412 MMGSEXTRADYNWS::
 51 72 A5 6C A5 52 DD 00C2 413 PUSHL R2
 74 A5 51 B1 00CA 414 SUBW3 PHDSW_PTCNTLCK(R5), PHDSW_PTCNTMAX(R5), R1 ; COUNT OF PAGE TABLES
 04 1B 00CE 415 ; WHICH ARE NOT LOCKED DOWN
 51 74 A5 3C 00D0 416 CMPW R1, PHDSW_WSFLUID(R5) ; MINIMIZE WITH FLUID COUNT
 51 74 A5 A0 00D4 417 BLEQU 10\$; BRANCH IF SMALLER
 51 6C A5 A0 00D8 418 MOVZWL PHDSW_WSFLUID(R5), R1 ; USE FLUID, IT IS SMALLER
 51 51 3C 00DC 419 10\$: ADDW PHDSW_WSFLUID(R5), R1 ; ADD IN FLUID FOR DATA & I STREAM PAGES
 00DF 420 ADDW PHDSW_PTCNTLCK(R5), R1 ; ADD IN LOCKED PAGE TABLE PAGES
 00DF 421 MOVZWL R1, R1 ; GET IT IN A LONGWORD
 00DF 422 : We now have count of WSLE's that must be reserved in dynamic portion of WSL
 52 18 A5 08 A5 A3 00DF 423 :
 52 B6 00E5 424 :
 50 A5 52 B1 00E7 425 SUBW3 PHDSW_WSLIST(R5), PHDSW_WSQUOTA(R5), R2
 04 1B 00EB 426 INCW R2 ; CALCULATE QUOTA
 52 50 A5 B0 00ED 427 CMPW R2, PHDSW_WSSIZE(R5) ; MINIMIZE WITH SIZE
 52 52 51 3C 00F1 428 BLEQU 20\$
 51 52 51 C3 00F4 429 MOVW PHDSW_WSSIZE(R5), R2 ; GET IT IN A LONGWORD
 00F8 430 20\$: MOVZWL R2, R2 ; TAKE AWAY THE RESERVED ONES
 00F8 431 SUBL3 R1, R2, R1
 00F8 432 : Remove non-dynamic portion of WSL from count
 00F8 433 :
 52 0E A5 08 A5 A3 00F8 434 :
 00F8 435 SUBW3 PHDSW_WSLIST(R5), PHDSW_WSDYN(R5), R2 ; GET COUNT OF LOCKED WS ENTRIES

52	52	3C	00FE	436	MOVZWL	R2,R2			
51	52	C2	0101	437	SUBL	R2,R1			
	02	14	0104	438	BGTR	30\$			
	51	D4	0106	439	CLRL	R1			
76	A5	51	80	0108	440	30\$:	MOVW	R1,PHDSW_EXTDYNWS(R5)	:GET IT IN A LONGWORD
	52	8E	D0	010C	441		MOVL	(SP)+,R2	:GET COUNT OF UNLOCKED ENTRIES
		05	010F	442			RSB		:BRANCH IF POSITIVE
			0110	443					:DON'T ALLOW A NEGATIVE EXTDYNWS COUNT
			0110	444					:SAVE IT IN PROCESS HEADER
					.END				

ACMSV_IMAGE	=	00000001		
CTLSGL_IWSPEAK		*****	X	02
CTLSGL_PHD		*****	XX	02
CTLSGL_WSPEAK		*****	X	02
EXESADJWSL		00000000	RG	02
EXESGL_ACMFLAGS		*****	X	02
IPLS_ASTDEL	=	00000002		
IPLS_SYNCH	=	00000008		
MMGSALCPHD		*****	X	02
MMGSEXTRADYNWS		000000C2	RG	03
MMGSFREWSLE		*****	X	03
MMGSSHRINKWS		00000039	RG	03
MMGSWSPEAKCHK		0000000D	RG	02
PAGCNT	=	00000004		
PCBSL_EFWM	=	0000004C		
PCBSW_GPGCNT	=	00000034		
PCBSW_PPGCNT	=	00000036		
PFNSGL_PHYPGCNT		*****	X	02
PHDSM_IWSPEAKCK	=	00000010		
PHDSM_WSPEAKCHK	=	00000004		
PHDSW_EXTDYNWS	=	00000076		
PHDSW_FLAGS	=	00000036		
PHDSW_PTCNTLCK	=	0000006C		
PHDSW_PTCNTMAX	=	00000072		
PHDSW_WSDYN	=	0000000E		
PHDSW_WSEXTENT	=	00000016		
PHDSW_WSFLUID	=	00000074		
PHDSW_WSLAST	=	00000012		
PHDSW_WSLIST	=	00000008		
PHDSW_WSNEXT	=	00000010		
PHDSW_WSQUOTA	=	00000018		
PHDSW_WSSIZE	=	00000050		
PRS_IPL	=	00000012		
SCHSGL_FREELIM		*****	X	02
SCHSGL_RESMASK		*****	X	03
SCHSGQ_MWAIT		*****	X	03
SCHSWAIT		*****	X	03
SGNSGL_MAXWSCNT		*****	X	02
SGNSGW_MINWSCNT		*****	X	03
SHRINK		00000000	R	
SSS_ACCVIO	=	0000000C		
SSS_NORMAL	=	00000001		
WSETLM	=	00000008		

! Psect synopsis !

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.09	00:00:00.68
Command processing	105	00:00:00.53	00:00:03.35
Pass 1	300	00:00:08.86	00:00:29.99
Symbol table sort	0	00:00:01.45	00:00:03.05
Pass 2	93	00:00:01.89	00:00:05.55
Symbol table output	6	00:00:00.07	00:00:00.61
Psect synopsis output	1	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	536	00:00:12.93	00:00:43.27

The working set limit was 1350 pages.

51408 bytes (101 pages) of virtual memory were used to buffer the intermediate code.

There were 50 pages of symbol table space allocated to hold 939 non-local and 25 local symbols.

444 source lines were read in Pass 1, producing 15 object records in Pass 2.

17 pages of virtual memory were used to define 16 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
\$255\$DUA2B:[SYS.OBJ]LIB.MLB;1	7
\$255\$DUA2B:[SYSLIB]STARLET.MLB;2	6
TOTALS (all libraries)	13

1020 GETS were required to define 13 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:SYSADJWSL/OBJ=OBJ\$:SYSADJWSL MSRC\$:\$SYSADJWSL/UPDATE=(ENH\$:\$SYSADJWSL)+EXECMLS/LIB

03B1 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

